Department of Computer Science

Chairperson: Nasri, Ahmad H.
Professor: Nasri, Ahmad H.
Associate Professors: Attie, Paul C.; Karam, Marcel R.; Safa, Haidar H.; Turkiyyah, George M.
Assistant Professor: Abu Salem, Fatima K.
Senior Lecturer: Jureidini, Wadi' N.
Lecturers: Awad, Mohamad M.; Haidar, May MA.; Keyrouz, Walid T.; Ladan, Mohamad I.
Instructors: Mukaddam, Wassim G.; Sidani-Bohsali, Hayat A.
Assistant Instructors: Akkari, Maha A.; Aoude, Loa K.; Arnaout, Huda A.; Bou Karam, Wajih, W.; Fatayri, Nour F.; Haddad, Daniel G.; Masri, Hani A.; Moussa, Bassam H.; Wehbe, Dana A.; Yassine, Zahra M.; Youssef, Kamal H.

The Department of Computer Science offers a program leading to the degree of Bachelor of Science (BS) in Computer Science. It also offers a program leading to the degree of Master of Science (MS) in Computer Science. For more information about the department visit its webpage: http://www.cs.aub.edu.lb/.

Undergraduate Program

BS in Computer Science

Mission Statement

In addition to the university requirements for graduate study in the Faculty of Arts and Sciences, the faculty is committed to cultivating lifelong learning that empowers the student to pursue continuous self-renewal through replenishing her/his knowledge base in an information technology driven age. The faculty offers excellent undergraduate programs leading to bachelor's degrees as well as graduate education leading to master's degrees in most fields. It expects to offer PhD degrees in selected fields in the future. The faculty builds its choices on areas of excellence as it develops its resources. The faculty considers research to be an essential element of faculty activity. Quality research is encouraged and supported to maintain the highest standard of education and to attract...
Department of Computer Science

outstanding faculty and students. The faculty also emphasizes the importance of high-quality teaching and continuously monitors teaching effectiveness. Student advising, another important faculty duty, is crucial to the proper running of its liberal arts program. To achieve excellence in education, the faculty continuously revises its curricula, enhances its different academic units by assessing their needs for equipment and research facilities, and concentrates on recruiting and maintaining prominent faculty in all professorial ranks. The active engagement of all its academic units in achieving its goals is vital for the lasting success of its three major areas of study: the humanities, the social sciences, and the basic sciences.

BS Degree in Computer Science

To graduate with a B.S. in computer science a student must finish:

University Requirements
- University language requirements (English 6 credits, Arabic 3 credits)
- University General Education requirements (Humanities 12 credits, Social Sciences 6 credits, Natural Sciences 6 credits, Quantitative Thought 3 credits).

Major Requirements
- Computer science: CMPS 200, CMPS 200L, CMPS 212, CMPS 212L, CMPS 253, CMPS 255, CMPS 256, CMPS 257, CMPS 258, CMPS 272, CMPS 277, CMPS 299, and nine additional credits in computer science courses numbered 220 and above
- Mathematics: MATH 201, MATH 211 (or CMPS 211), and one Math course to be chosen from MATH 218, MATH 219, STAT 230, STAT 233, and MATH 261.
- Sciences: Physics 228, 228L
- Free elective: one course numbered 200 and above from outside the department.

All prospective computer science majors are expected to complete CMPS 200, MATH 201, MATH 211 or CMPS 211, CMPS 212, CMPS 212L, and CMPS 255 in the sophomore year. Computer science majors are expected to complete CMPS 253, CMPS 256, CMPS 257, and CMPS 258 in the junior year, and maintain an average grade of at least 70 in computer science courses. Finally, students are encouraged to take ACCT 210 and PHIL 211 among their general/Free elective courses.

A minor in computer science requires 18 credits: CMPS 200, CMPS 212, CMPS 255, and nine additional credits in computer science courses (CMPS) numbered 211 or above. A minimum of 9 credits must be taken in the department. [Note: This minor is not open to EECE students.]

A concentration in a specific area in computer science consists of 12 credits: 9 credits in computer science to be chosen from the list of the corresponding area, and CMPS 299 (which is to be carried out in the same area.) The first concentration that will be introduced is in Graphics and Multimedia. The list of courses in this area includes: CMPS 220, CMPS 259, CMPS 285, CMPS 286, CMPS 289, CMPS 290, CMPS 388, CMPS 385, CMPS 387, CMPS 396 and CMPS 395.
Sample Study Plan
A typical study plan could have the following distribution of CMPS courses:

First Year
First Semester: CMPS 200, CMPS 200L, CMPS 211, MATH 201
Second Semester: CMPS 212, CMPS 212L, CMPS 255, CMPS elective

Second Year
First Semester: CMPS 253, CMPS 256, MATH course
Second Semester: CMPS 257, CMPS 258, CMPS elective

Third Year
First Semester: CMPS 277, CMPS 272
Second Semester: CMPS 299, CMPS elective

Undergraduate Courses

CMPS 200  Introduction to Programming  3.3; 4 cr.
An introduction to a disciplined approach to computer programming and problem solving, utilizing a block-structured high level language, with an emphasis on procedural abstraction and good programming style. This course covers the basic repetition and selection constructs, procedures and functions, parameter passing, and scope of variables. Note: If EECE 230 is completed, students can get credit for only one of CMPS 200 or EECE 230. Each semester.

CMPS 200L  Introduction to Programming Lab.  1.3; 1 cr.
This laboratory course focuses on programming and developing application programs that cover the lab activities of the CMPS 200 course. It introduces the students to a development environment to apply the various programming concepts and techniques gained in CMPS 200 classes. Weekly programming assignments are a key component of the course. In addition to lab works, students are expected to attend a weekly recitation session in which the topics of the next lab work session is discussed. Corequisite: CMPS 200. Each semester.

CMPS 206  Computers and Programming for the Arts  2.3; 3 cr.
An introduction to computers and an illustration of their use. Common applications are considered in word processing, spreadsheets, and database systems. This course also includes an introduction to the Internet and the World Wide Web. This course is meant to be a computer literacy course open to Arts students only. No credit is given to computer science majors. Students can get credit for only one of CMPS 206, CMPS 209, or EDUC 219. Annually.

CMPS 209  Computers and Programming for the Sciences  2.3; 3 cr.
A computer literacy course covering all the topics in CMPS 206. Additionally, this course provides an introduction to programming using Visual Basic or a similar language. No credit is given for computer science majors. Students can get credit for only one of CMPS 206, CMPS 209, or EDUC 219. Each semester.
CMPS 210  Digital Imaging  2.3, 3 cr.
A course that introduces the fundamental concepts, terminology and techniques needed for the creation of digital images with emphasis on Digital Artwork. Topics include: Pixel-based imaging, Vector-based imaging, Digital image acquisition, Digital image Creation and Manipulation, Colors in computer graphics, 3D objects modeling, shading, lighting and texture. It also focuses on the technical practices, aesthetic and ethical issues of digitally created images. Annually.

CMPS 211  Discrete Structures  3.0; 3 cr.
Logical reasoning, sets, relations and functions; mathematical induction, counting, and simple finite probability theory; molecular arithmetic in different bases; recurrence relations and difference equations; truth tables and switching circuits; graphs and trees; strings and languages. This course is equivalent to Math 211. Annually.

CMPS 212  Intermediate Programming with Data Structures  3.3; 4 cr.
A continuation of CMPS 200, this course consolidates algorithm design and programming techniques, emphasizing large programs. This course also provides a detailed study of data structures and data abstraction, and an introduction to complexity considerations and program verification. Note: If EECE 330 is completed, students can get credit for only one of CMPS 212 or EECE 330. Prerequisite: 200 and CMPS 200L or EECE 230. Corequisite with CMPS 212L. Each semester.

CMPS 212L  Introduction to Programming with Data Structures Lab.  1.3; 1 cr.
This course consists of the programming lab activities of the CMPS 212 course. It focuses on the role of Abstract Data Types and Data Structures in relatively large programs. The use of existing data structures, and the implementation of new Abstract Data Types are also considered. Weekly programming assignments are a key component of the course. In addition to lab works, students are expected to attend a weekly recitation session in which the topics of the next lab work session are discussed. Prerequisites: (CMPS 200 or EECE 230) and CMPS 200L. Corequisite CMPS 212. Each semester.

CMPS 220  Foundations of Digital Media  3.0; 3 cr.
This course introduces the process of digitizing media such as images, audio, and video, and the use of software tools used to manipulate digital media. In addition, compression, generation and other useful introductory topics will be presented. The course will be practiced with hands on approach that will highlight the fundamentals of digital media. This course assumes basic knowledge in Java or first course in programming. Prerequisite: Junior standing or consent of instructor. Annually.

CMPS 251  Numerical Computing  3.1; 3 cr.
Techniques of numerical analysis: number representations and round-off errors, root finding, approximation of functions, integration, solving initial value problems, Monte-Carlo methods. Implementations and analysis of the algorithms are stressed. Projects using MATLAB or a similar tool are assigned. Prerequisites: (CMPS 200 or EECE 230) and MATH 201. This course is equivalent to MATH 251. Annually.

CMPS 253  Software Engineering I  3.0; 3 cr.
A course that introduces the fundamentals of software engineering, with emphasis on the requirements elicitation and specification, and analysis and design phases of the software life cycle. Specifications are given as a set of operations (with pre- and post-conditions), and using a generic data model, and the design as a module dependency diagram where both data and procedural decomposition are emphasized. The course also introduces verification and testing of a design with respect to its specification, and the use of modularity and decomposition to ensure tractability of the verification. Students will apply the concepts learned by to developing a software system. Prerequisites: CMPS 212 and CMPS 212L or EECE 330. Annually.
CMPS 255  Computer Architecture 3.0; 3 cr.
A structured overview of the fundamentals of designing digital computer systems. Topics covered include digital logic and systems, machine level representation of data, assembly level machine organization, memory system organization and architecture, CPU implementation and virtual machines, and exposure to one or more micro/mini architectures. Prerequisite or co-requisite: CMPS 212 and 212L or EECE 330. Annually.

CMPS 256  Advanced Algorithms and Data Structures 3.0; 3 cr.
A systematic study of algorithms and their complexity. Topics include techniques for designing efficient computer algorithms, proving their correctness, and analyzing their complexity; as well as advanced searching, sorting, selection, graph and matrix algorithms. Prerequisite: CMPS 212 and 212L or EECE 330. Annually.

CMPS 257  Theory of Computation 3.0; 3 cr.
A course that covers basic theoretical principles embodied in automata and grammars. Topics include regular expressions, finite automata, context-free grammars and parsing, pushdown automata, closure properties, Turing machines, Church's thesis, reductions and decidability. This course also provides a quick introduction to complexity theory. Prerequisites: (MATH 211 or CMPS 211) and (CMPS 212 and 212L or EECE 330). Annually.

CMPS 258  Programming Languages 3.0; 3 cr.
A course on the principles and programming styles that govern the design and implementation of contemporary programming languages, a history and overview of programming languages, fundamental issues in language design, and an introduction to language translation. This course focuses on design issues in imperative, object-oriented, functional, and rule-based paradigms. This last paradigm will be used to introduce intelligent systems issues. Languages such as C, C++, Haskell, and Prolog are used to illustrate key concepts. Prerequisite: CMPS 212 and 212L or EECE 330. Annually.

CMPS 259  Multimedia Programming 3.0; 3 cr.
This course introduces how to design databases and to use authoring tools’ built-in scripting languages to reduce the overhead for a multimedia product. Topics include: application of the scripting language to accomplish other sophisticated effects, and the use the scripting languages built into authoring tools to extend their ability to create multimedia products with increased efficiency and capabilities. Prerequisites: CMPS 212, 212L and CMPS 220. Annually.

CMPS 272  Operating Systems 3.0; 3 cr.
An overview of operating systems and net-centric computing. Topics include operating system principles, scheduling and resource management, virtual memory, file systems, concurrent processing and synchronization, security and protections, the Internet, network structures, distributed operating systems, and Web technologies and operating systems (URL, HTML, HTTP, applets). A case study of a contemporary operating system like UNIX accompanies the course. Prerequisites: (CMPS 255 or EECE 321) and (CMPS 256 or EECE 330). Each semester.

CMPS 273  Network Programming 3.0; 3 cr.
This course focuses on the programming aspects of networking protocols. Topics include: designing and building programming applications that use computer networks, fundamental concepts required to build iterative and concurrent client/server networking applications using sockets. Then it moves to explain low level networking programming and other advanced socket topics. The course also presents the emerging peer-to-peer computing along with some tools needed to develop P2P applications. Prerequisites: 272 or consent of instructor. Annually.
CMPS 274  Compiler Construction  3.0; 3 cr.
A course that covers syntax specifications of programming languages, parsing theory, top-down and bottom-up parsing, parser generators, syntax-directed code generation, symbol table organization and management, dynamic storage allocation, code optimization, dataflow analysis, and register allocation. Prerequisites: CMPS 255, CMPS 258 and CMPS 257. Biennially.

CMPS 277  Database Systems  3.0; 3 cr.
An overview of the nature and purposes of database systems and an introduction to data modeling: entity relationship model, relational model with relational algebra, relational calculus and SQL; integrity constraints; file organization and index files; normalization. Prerequisite: CMPS 256 or EECE 330. Annually.

CMPS 278  Web Programming and Design  3.0; 3 cr.
This course introduces the exciting world of WWW, the fundamentals needed to program on the Internet, and the state of the art technologies used in designing and developing rich multi-tiered web based applications. It presents the basics of client-side/server-side web programming and all the skills and tools needed to create dynamic Web-based applications. It provides in-depth coverage of various markup languages (XHTML, Dynamic HTML and XML) and their associated cascading style sheets, several client side and server side scripting languages (JavaScript, PHP) in addition to AJAX-enabled rich Internet applications, client-side technologies, web services, Web Servers, and multi-tiered applications using relational database systems. Prerequisites: CMPS 277 or consent of instructor. Annually

CMPS 281  Numerical Linear Algebra  3.0; 3 cr.
A course on direct and interactive methods for solving general and special systems of linear equations, covering LU decomposition, Choleski decomposition, nested dissection, marching algorithms; Jacobi, Gauss-Seidel, successive over-relaxation, alternating directions, and conjugate gradient iterative methods. This course is equivalent to MATH 281. Prerequisites: (MATH 218 or 219) and (MATH 251 or CMPS 211). Annually.

CMPS 282  Software Engineering  3.0; 3 cr.
A course on software engineering dealing with large systems, including use of APIs, management of software teams, and software testing and validation; data flow concepts and decision tables; conditions and decision variables; design of output and input forms; files and database development; on-line and distributed environments; system documentation; and system implementation. Students are expected to complete a project in which they integrate their knowledge of the undergraduate computer science curriculum by implementing a significant software system in team work. Restricted to students who joined the program before October 2010. Prerequisite: CMPS 253. Annually

CMPS 283  The Logic of Programming  3.0; 3 cr.
A course on computer programming as a rigorous mathematical discipline. Topics include sentential logic, predicate logic, expressions and commands, pre/post-conditions, assignment, repetition, invariant predicates, function predicates, modules, data structures, and concurrency. Prerequisites: (CMPS 212 or EECE 330) and CMPS 211. Biennially.

CMPS 284  Computer Networks  3.0; 3 cr.
An introduction to basic data communication, network architecture, protocols, local area networks, and wide area networks. Special emphasis is placed on the TCP/IP protocol suite. The BSD socket library is presented. Prerequisite: CMPS 255 or EECE 321. Annually.
CMPS 285  Computer Graphics  3.0; 3 cr.
A course that covers the practice of, and underlying mathematical foundation for, interactive graphics programming. Topics include basic graphics systems, graphics primitives and attributes, windows and viewports, clipping, geometric transformations, color systems, 2D texture mapping, and introduction to 3D graphics. Programming in OpenGL will be used. Prerequisite: CMPS 212 or EECE 330. Annually.

CMPS 286  Computer-Aided Geometric Design  3.0; 3 cr.
A course that discusses the representation of free-form curves and surfaces in modeling objects by computers, including curve approximation and interpolation, spline curves (Bezier and B-splines), visual smoothness of curves, geometric continuity, parameterization of curves, introduction to surface interpolation and approximation, and spline surfaces (Bezier and B-splines). Prerequisite: CMPS 212 or EECE 330. Biennially.

CMPS 287  Artificial Intelligence  3.0; 3 cr.
An introduction to the principles and techniques that enable computers to behave intelligently. This course covers basic problem solving methods, knowledge representation, reasoning methods, learning from samples and from experience, expert systems and knowledge acquisition, machine learning, and neural networks. Several projects are given, some of which are in Prolog. Prerequisites: CMPS 256 and 258. Annually.

CMPS 288  Internals of Database Management Systems  3.0; 3 cr.
A course on the internals of database management systems, especially relational DBMS. Topics include query processing and optimization, transaction processing, concurrency control, recovery, distributed transactions, database security, client-server, multi-tier architectures, and web deployed database systems. Prerequisite: CMPS 277. Annually.

CMPS 289  Human Computer Interaction  3.0; 3 cr.

CMPS 290  Multimedia Design  3.0; 3 cr.
This course introduces the theoretical and practical aspects of the principles governing the design in multimedia, computer-enhanced and computer generated environments. In particular, it covers the design process for multimedia products, including the development life cycle and basic interface design issues. Each student will design a significant multimedia product and use a typical authoring environment to create a working prototype. Prerequisite CMPS 259. Annually.

CMPS 296  Computer Science Tutorial  1–3 cr.
Prerequisite: Senior standing.

CMPS 297  Special Topics in Computer Science  1–3 cr.
A course on selected topics which change according to the interests of the instructors and/or students. Topics are chosen from state-of-the-art innovations in software and computer information systems. Prerequisite: Consent of instructor. Annually.
CMPS 299  
Graduation Project in Computer Science  
3 cr.
A course to enhance students’ skills with practical experience giving them the opportunity to integrate knowledge accumulated in different courses. In this course, students must deliver products, in their area of emphasis/concentration, which pass through the design, analysis, implementation, testing, and evaluation stages. Prerequisite: Senior standing. Annually.

41 Credits in Computer Science

<table>
<thead>
<tr>
<th>Modes of Analysis</th>
<th>English and Arabic (9)</th>
<th>Humanities (12)</th>
<th>Social Sciences (6)</th>
<th>Natural Sciences (10)</th>
<th>Quantitative Thought (32+9+9+3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture Courses</td>
<td>(9+12+6+10+53)</td>
<td>Required credits in the humanities: 12 credits including 6 credits from CVSP (see pp. 158-63)</td>
<td>Required Courses (6)</td>
<td>Required natural science courses (6)</td>
<td>1. Required CMPS courses (32): CMPS 200(3)+200L(1), CMPS 212 (3)+212L (1), CMPS 253(3), CMPS 255(3), CMPS 256(3), CMPS 257(3), CMPS 258(3), CMPS 272(3), CMPS 277(3), CMPS 299(3)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. Required Courses (usually 6): ENGL 203(3), CMPS 204(3), as determined by placement</td>
<td></td>
<td>2. Required CMPS electives (9): to be chosen from CMPS courses above 220</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Required natural science courses (6): MATH 201(3), MATH 211(3) or CMPS 211) and one Math course to be chosen from MATH 218, MATH 219, STAT 230, STAT 233, and MATH 261. Note: since MATH 251 is equivalent to CMPS 251, it cannot count as both a computer science elective and mathematics elective</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. One free elective (3) numbered 200 and above from outside the department</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Seminar (0)
Laboratory (3) PHYS 228L CMPS 2001, 2122 (4 hrs/week)
Research Project (0)

1 Natural science courses are numbered 200 and above and drawn from biology, chemistry, geology or physics, open to science students
2 CMPS 200L and CMPS 212L are each a 1-credit course with 1 recitation hour, and 3 lab hours per week.
3 CMPS 211 and CMPS 251 are 3-credit courses with 3 lecture hours and 1 recitation hour